

IN THE SPECIFICATION

Please amend the portions of the Specification identified below to read as indicated herein.

Paragraph starting at page 6, line 16:

The code generation system 119 utilizes markup files including, for example, Extensible Markup Language (XML) documents. XML is a subset of Standard Generalized Markup Language (SGML) which is used for documents with structured information. ~~XML is described in XML 1.0, found at the following web site: <http://www.w3.org/TR/REC-xml>.~~ XML is a set of rules or guidelines for designing text formats for structured data using tags. ~~Additional detail may be found at the following web site: [http://www.w3.org/XML/1999/XML in 10 points](http://www.w3.org/XML/1999/XML-in-10-points).~~ XML is a new specification that is quickly gaining popularity for creating what are termed "XML documents" which may be structured by various elements or attributes (e.g., title, data, author, etc.). Once a document is structured in this manner, a structured search may be performed based on element or attribute values (or content).

Paragraph starting at page 7, line 4:

~~In particular, Extensible Markup Language (XML) is a subset of Standard Generalized Markup Language (SGML).~~ XML works in conjunction with Extensible Stylesheet Language Transformation, (XSLT) and Extensible Markup Language Path (XPath). XML may also work in conjunction with a Document Object Model (DOM) or Namespace.

Paragraph starting at page 7, line 8:

~~Extensible Markup Language (XML) is a subset of Standard Generalized Markup Language (SGML). XML is described in XML 1.0, found at the following web site:~~

~~http://www.w3.org/TR/REC-xml. Extensible Markup Language (XML) is a set of rules or guidelines for designing text formats for structured data using tags. Additional detail may be found at the following web site: http://www.w3.org/XML/1999/XML in 10 points.~~ For interoperability, domain-specific tags called a vocabulary can be standardized using a Document Type Definition, so that applications in that domain understand the meaning of the tags.

Paragraph starting at page 7, line 15:

Extensible Style Language Transformer or XSLT is a language for transforming XML documents into other XML documents. The XSLT specification defines the syntax and semantics of the XSLT language. XSLT-defined elements are distinguished by belonging to a specific XML namespace, which is referred to as the XSLT namespace. A transformation expressed in XSLT describes rules for transforming a source tree into a result tree. ~~Further detail about XSLT may be found at http://www.w3.org/TR/xslt.~~

Paragraph starting at page 7, line 21:

XML Path or XPath addresses parts of an XML document. XPath gets its name from its use of a path notation as in URLs for navigating through the hierarchical structure of an XML document. ~~Further detail about XML path may be found at http://www.w3.org/TR/xpath.~~

Paragraph starting at page 7, line 24:

A Document Object Model (DOM) is a standard set of function calls for manipulating XML files from a programming language. ~~Additional detail may be found at the following web site: http://www.w3.org/TR/REC-DOM-Level-1/.~~

Paragraph starting at page 9, line 19:

The system description in the markup language file 200 is input into a code generator 230. The code generator 230 reads the markup language file 200, creates objects based on the system description, and outputs a structure of a database 240 and one or more programs 250 (e.g., a search engine program) for storing and retrieving data in the database. In an alternative embodiment, the code generation system may output a database structure which forms a digital library 107. In an additional embodiment, a digital library 107 may include a single database. In another embodiment, the digital library 107 may include a collection of databases. In addition, a search engine program ~~260~~ 250 output by the code generator ~~240~~ 230 may locate objects or object data in the custom database or digital library 107.

Paragraph starting at page 9, line 28:

In an alternative embodiment, the same program may both create the database structure 240 and serve as a search engine 250 for that custom database even though FIG. 2 illustrates separate programs for the database structure 240 ~~an~~ and search engine 250. The manner in which these components are utilized by the code generation system 119 is explained in further detail with reference to FIG. 3.

Paragraph starting at page 10, line 4:

FIG. 3 is a flow diagram illustrating the technique of the code generation system 119. In block 300, the code generator 230 receives a markup file containing a system description 200 of the structure of the custom database to be created. The system description 200 of the custom database defines the mapping of the abstract digital asset to the physical representation in a database. In other words, the system description 200 defines the structure of the custom database to be created through various objects. For example, the system description 200 may include, a set of data that is to be stored in a database, how that data is to be stored, how the data is linked or associated with other data, and any other information relating to how the data should be stored and/or

processed. Those skilled in the art will recognize that numerous other object structures and associations may be represented in the system description 200.

Paragraph starting at page 10, line 15:

A markup language file stores the system description 200. For example, the system description 200 may be entered into the markup of an Extensible Markup Language (XML) document. In an alternative embodiment of the present invention, the system description 200 may be entered into a Java document. The system description 200 may be entered into a markup file using a text editor 210 or by inputting data through a graphical user interface (GUI) 220. Those skilled in the art will recognize that the system description may be entered using other data entry systems. An example of one embodiment of the present invention includes the following XML document with a system description 200:

Paragraph starting at Page 14, line 25:

Different system descriptions 200 may describe the same custom database, and thus, different system descriptions may generate the same database structure 240. In other words, the database structure 240 may be transparent to changes in the system description 200 such that the same database structure 240 is generated with different system descriptions 200.

Paragraph starting at page 14, line 29:

In block 330, the code generator 230 generates a program, such as a search engine program 250, to locate data in the database structure 240, ~~such as a search engine 250~~. When a search query is executed against the ~~custom~~ database structure 240 by a search engine program ~~260~~ 250, the search engine program ~~260~~ 250 may refer to the index created ~~by~~ for the database ~~program 250~~ structure 240 when the objects were ~~stored~~ stored, and retrieve the objects or object data requested by the query. As objects are

modified or as new ~~object~~ objects are added or deleted by, for example, a different system description 200, ~~then the database~~ the search engine program 250 may store or delete these objects and index them accordingly. The search engine program 250 may execute queries against the ~~custom~~ database structure 240 using search engines executed in Java, Hyper Text Markup Language (HTML) or other high level languages. In addition, the search engine program ~~260~~ 250 may be a text search engine. Those skilled in the art will recognize that the code generator 230 may generate search engine programs ~~260~~ 250 executable using various language formats.